

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑪ 公開特許公報(A) 平4-64129

⑫ Int. Cl.

識別記号

庁内整理番号

⑬ 公開 平成4年(1992)2月28日

G 06 F 9/06

4 5 0 J

7927-5B

4 5 0 L

7927-5B

12/14

3 2 0 E

7165-5B

審査請求 有 発明の数 1 (全10頁)

⑭ 発明の名称 ソフトウェア管理方式

⑮ 特 願 平2-211406

⑯ 出 願 昭58(1983)10月5日

⑰ 特 願 昭58-186100の分割

⑱ 発 明 者 森 亮 一 東京都文京区白山1-24-12

⑲ 出 願 人 森 亮 一 東京都文京区白山1-24-12

⑳ 代 理 人 弁理士 長谷川 文廣 外1名

明 細 書

1. 発明の名称

ソフトウェア管理方式

2. 特許請求の範囲

(1) ソフトウェアに対応したソフトウェア固有データを格納したソフトウェア記憶手段、

ソフトウェアの利用者に対応した利用者固有データを格納した利用者固有データ記憶手段、

上記ソフトウェア固有データと上記利用者固有データ各々の少なくとも一部のデータを用いてソフトウェアの利用可否を判定する利用可否判定手段、及び、

該利用可否判定手段が利用可と判定した時、ソフトウェア固有データに対応したデータを格納するための利用ソフトウェア履歴格納手段とを備え、

該利用ソフトウェア履歴格納手段の格納内容は上記ソフトウェア固有データに対応して管理

されていることを特徴とするソフトウェア管理方式。

(2) ソフトウェア固有データとしてソフトウェア権利者識別符号を備え、

利用者固有データとしてソフトウェア利用条件を備え、

利用可否判定手段が利用可と判定した時、ソフトウェア権利者識別符号を利用ソフトウェア履歴格納手段に格納し、ソフトウェア権利者対応にそのソフトウェア利用状況を記憶可能としたことを特徴とする特許請求の範囲第1項記載のソフトウェア管理方式。

3. 発明の詳細な説明

(a) 技術分野

本発明は、無形財産であるコンピュータ・プログラムやビデオ等のソフトウェアの管理方式に関し、特にソフトウェアの利用状況をソフトウェア権利者などによって把握できるようにしたソフトウェア管理方式である。

## (3) 発明の背景

近年、データ処理システムの発達と共に種々の有償プログラムが販売されるようになったが、その保護は不完全であり、プログラムの不正利用も多い。この不正利用を防ぐために、特許法、著作権法更には特別法などが検討されているが、ソフトウェアという概念最近に至って生じた代物に対しては、何れをとってもその保護ということについては、不完全である。これは、単に無形財産/有形財産の問題でなく、権利者がその取引を知っているか否かの問題であり、その点をうまく解決できるか否かが、この保護を行えるか否かの鍵となる。

## (4) 従来の問題点

ソフトウェア、例えば、コンピュータ・プログラムでは一度利用者の手に渡るとそれ以降、どの様にそのプログラムが利用されているかを権利者は知らず、利用は全く野放しの状態である。この状態は本来利用しただけ支払うという取引形態

のみ該ソフトウェアを実行させるようにして、利用出来ないようにする手法等はあった。(複写して取得されたソフトウェアは複写不一致になるので実行出来ない。)

しかしながら、その複写管理は大変だし、ましてや、有償ソフトウェアを店頭にて販売しようとする、実質上その様な複写による方法は採用不可能であった。

## (5) 発明の目的

従って、本発明の目的は、ソフトウェアの利用状況を権利者が把握できる様にあることにある。

## (6) 発明の構成

この目的は、ソフトウェアに対応したソフトウェア固有データを格納したソフトウェア記憶手段、ソフトウェアの利用者に対応した利用者固有データを格納した利用者固有データ記憶手段、上記ソフトウェア固有データと上記利用者固有データ各々の少なくとも一部のデータを用いてソフトウ

ェをコンピュータ・プログラムについてはとっていないことを意味し、一種の食い逃げが許されていることに等しい。この様な体制では、プログラムの利用料金は高くなってしまっただけでなく、粗悪なソフトウェアの氾濫を許し、極めて細かなサービスを行うソフトウェアの提供意欲が減退してしまうものである。従って、若しコンピュータ・プログラムについてプログラム権利者に極めて細かに保護を行うシステムを社会が提供すれば、該社会ではより良いプログラムの発生が促され、社会がより発展することになる。これは、コンピュータ・プログラムの占める重要性が増す将来において、極めて大事な問題である。

コンピュータ・プログラムを含む複写という従来における不正行為については、それを防制するための手段として、電子計算機等のハードウェアに付与されている複写を用い、ソフトウェア自体にも該複写に相当する符号を付与し、プログラムの実行に際しては、ハードウェア複写とソフトウェアに付与された複写とを照合し、一致した場合

の利用可否を判定する利用可否判定手段、及び、該利用可否判定手段が利用可と判定した時、ソフトウェア固有データに対応したデータを格納するための利用ソフトウェア履歴格納手段とを備え、該利用ソフトウェア履歴格納手段の格納内容は上記ソフトウェア固有データに対応して管理されるようにすることで達成される。

## (7) 発明の実施例

第1図は、本発明のソフトウェア・サービス・システム(SSS)の概念を説明するための図である。図において、Pはソフトウェア権利者、PPは有償ソフトウェア(Program Product)、PIDはソフトウェア固有データ、USERIDは利用者固有データ、CHECKは利用可否判定手段、SHは利用ソフトウェア履歴格納手段、DPSはデータ処理システム、SSUはソフトウェア・サービス・ユニットである。

ソフトウェア権利者Pはソフトウェア固有データPIDを含めた形で有償ソフトウェアPPを

提供する。ユーザのデータ処理システムDPSでは、その有償ソフトウェアPPを利用する際には、必ずソフトウェア・サービス・ユニットSSUを経由する様に構成されており、そのソフトウェア・サービス・ユニットSSUには利用者固有データUSERIDが格納されている。有償ソフトウェアPPの利用要求が生じると、利用可否判定手段CHECKは、利用者固有データUSERIDに基づいて、指定されたソフトウェアの利用可能性をチェックする。例えば、資格の有無、その使用料金がそのユーザに許容されている利用可能金額以内または利用可能時間以内か否かなどのチェックである。もし、OKであればその旨が図示されるデータ処理システムDPSのオペレーティング・システム（以下OSと略す。）に通知されるとともに、利用ソフトウェア履歴格納手段SHに利用者固有データUSERID及びソフトウェア固有データPIDが格納される。もし、OKでなければ、利用できない旨の返答がデータ処理システムDPSのOSになされる。

ムを管理する管理組合のSS協会、3-1~3-nは有償プログラムに関する実施があった時にその対価を受領する権利のあるプログラム権利者（複数権利者であった時は、その配分率情報も記入しておく。）、4a~4zは有償プログラムで、4aと4bはプログラム権利者3-1が作成したもの、4yと4zはプログラム権利者3-nが作成したもの、5と6は有償プログラム、7は中央処理システム、8はソフトウェア・サービス・ユニット(SSU)、9は利用者固有データ格納手段の一部で利用者識別符号を格納するユーザ識別符号メモリ、10は有償プログラムの利用可否を判定する利用可否判定手段である利用可否判定処理部、11は利用者固有データ格納手段の他の一部で通貨記号（\$、¥又は£など）を含めた利用可能金額メモリ及び利用ソフトウェア履歴格納手段を含む利用プログラム履歴メモリ、12は利用プログラム履歴メモリ11の内容の入出力を処理する入出力処理部、13は中央処理システム7におけるOS部、14は磁気ディスク装置、15は

この利用ソフトウェア履歴格納手段SHの内容は、ソフトウェア権利者Pが閲覧可能であるとともに、該利用ソフトウェア履歴格納手段SHの内容により、たとえばソフトウェア利用料金をソフトウェア権利者別に求めることが可能となる。

本発明によれば、有償ソフトウェアPPについての利用状況がソフトウェア権利者Pに把握可能となるので、その利用状況に見合った割合で利用料金の徴収が可能となり、ソフトウェアの入手費用を安くしておくことで、利用者であるユーザは種々のソフトウェアを試用可能となり、且つ良いソフトウェアだけを利用してゆくことが可能となる。これにより、粗悪ソフトウェアは利用されなくなるし、良いソフトウェアはより頻繁に利用されるようになるので、ソフトウェア権利者Pは益々良いソフトウェアを提供しようと競い合うことになる。

第2図は、本発明が適用される実施例である。図において、1はユーザにおけるデータ処理システム、2は有償ソフトウェアである有償プログラ

プリンタ、16はキーボード、17は表示装置である。

さて、第2図のシステムにおいては、まずプログラムを作成し、そのプログラムを有償プログラムとして、一般に利用させ、且つその利用料金を確実に回収しようとするプログラム権利者は、SS協会へ行き、その利用料金の振り込み先口座名を登録し、プログラム権利者識別符号（以下プログラム権利者IDと称す。）を受け取る。協会では、各プログラム権利者に対して、異なるプログラム権利者IDを付与する。この行為は例えば、第2図の窓口Wで行われ、管理側NTで管理される。プログラム権利者3-1は例えばP1というプログラム権利者IDが付与されているものとする。

プログラム権利者P1は、有償プログラム4aと4bを作成したとする。このプログラムに上記プログラム権利者IDであるP1、プログラムのバージョン（版）及び利用料金等のプログラム固有データをプログラム権利者は付与し、カセッ

ト型の磁気テープ等の記憶媒体に格納し、店頭販売若しくは通信販売等でプログラムの配布を行う。勿論オンラインで直接経由での配布も可能である。

一方、ユーザは種々の雑誌や店頭等で、自分の目的にあったプログラムを探す。そして、目的のプログラムがSS協会のメンバーが作成したものであり、第2図の4aだったとすると、該プログラムを最低価格または無料で取得する。ユーザのデータ処理システム1にはそのデータ処理システムにSS協会2の登録を受けたプログラムを処理するための機構として、ソフトウェア・サービス・ユニット8がインストールされていなければならない。

このソフトウェア・サービス・ユニット8には、ユーザ識別符号メモリ9からのデータと、利用される有償プログラムに付与されたプログラム固有データ及び利用プログラム履歴メモリの内容をチェックし、利用可否を判断する利用可否判定処理部10、及び、利用プログラム履歴メモリ11の内容に関し、プログラム利用履歴を出力したり、

利用可能金額を更新したりする入出力処理部12を備えている。この入出力処理部12は、各ユーザにより種々の形態をとることが可能であり、オンラインで直接離れた場所にあるファイルに転送するものや、外部記憶媒体、例えば磁気カードやマイクロ・プロセッサを中に備えたICカードなどでも良い。その利用形態により、種々のソフトウェア・サービス・ユニット8が存在し得る。

データ処理システム1が有償プログラム5を利用する時、つまり、ジョブ制御言語の解釈結果で該ソフトウェアの利用をOS部13が検出すると、その旨がソフトウェア・サービス・ユニット8に通知される。これは、特権割り込みと同等の制御で良い。ソフトウェア・サービス・ユニット8は利用プログラム履歴メモリ11中の利用可能金額を読み出すと共に、該有償プログラム5内にあるプログラム固有データからの利用料金を求める。この利用料金を上記利用可能金額から減じる処理が次になされる。その結果が、負だったら、そのプログラムは利用不可である為、その旨がOS部

13に通知され、例えば、システムメッセージ「プログラム4aは残高が無いので利用出来ません」が表示装置17に表示される。(後述の如く、負の場合でも条件付で利用可とすることも可能である。)

逆に、上記利用可能金額から利用料金を減じた結果が負でなかったら、該減じた値を利用プログラム履歴メモリの利用可能金額の記憶領域へ格納するとともに、そのプログラムに付与されているプログラム権利者IDである、P1とその利用料金、及びユーザ識別符号メモリにあるユーザ識別符号を同じく利用プログラム履歴メモリ11にその利用明細として格納する。勿論、そのプログラムを識別するためのコードも料金支払いに不必要な情報ではあるがプログラム利用状況を把握するためのデータとして利用価値があり、利用プログラム履歴メモリ11に格納される。そして、OS部18に対して、処理の執行を促す。

以下、同様に処理が進められる。従って、利用プログラム履歴メモリ11には常に残高として利

用可能金額が記されていると同時に、プログラム権利者ID毎にどのユーザ識別符号のユーザが幾ら支払うべきかが記されていることになる。

本システムにおいては、利用プログラム履歴メモリ11の利用明細データが格納される領域は所定量または所定額となっており、該限界を越えた場合には利用プログラム履歴メモリ11の内容を空にする処理が必要となる構成してある。高額の情况等、途中でその利用状況をSS協会へ報告させる手段を設けてある。この利用プログラム履歴メモリ11の残高メモリに対しても、入出力処理部によりその額を外部より増加可能としている。銀行オンライン方式と同様の構成または上述のICカードをSS協会より所定の料金でユーザに購入させ、そのICカード内に記憶されている料金を増加させる構成をとり得る。

プログラム固有データとして特約データを含めることが本発明で可能となる。この場合、上記利用可否判定処理部10はプログラム固有データの中に特約データが記入されていると利用者固有デ

一タに利用特約コードが有る場合のみ利用可と判定する。これは、プログラム権利者が特に特定プログラムについては、その利用者を把握しておきたい時に便利である。利用特約コードを利用者が得るためには、利用者はSS協会と特約をする必要があり、特約をするためには当然のこととして利用者の名前などがSS協会に登録されるので、プログラム権利者がその名前等を把握するのは容易である。また、その特約内容として報告義務を課する様に、例えば、SS協会へ報告するのをやめ、下記の如き報告に対する所定割引きによってもうけることよりは、プログラム作成者へお金支払われることがないようにすることを選ぶというライバルへの不正行為のような、ダンピング類似行為を防止したりすることも可能となる。

本システムでは、利用プログラム履歴メモリ11の中の利用明細をSS協会が知ることがその利用されたプログラムのプログラム権利者へ料金支払いをする上で必須である。実現方法として、利用明細の報告があった場合、その内の例えば

10%をユーザに返済するという手段を講じてある。従って、ユーザとしては、報告した方が自分の利益になるので、殆どは報告をすることになり、プログラム権利者は確実にその利用量を回収することが可能となる。第2図では、報告を受けたSS協会2は、プログラム権利者IDで管理簿NTを参照し、その対応した振り込み先口座へその料金を振り込むという処理を行う。

ICカードを用いた場合でも全く同じであり、用済みICカード及び換金したいICカードは同じくSS協会へ届けられると考えられている。即ち、SS協会では、ICカード持参者または送付者に対して、利用金額に応じ、ICカード返済還元金額を決定してある。例えば、利用金額累計の10%及び預金としての利用可能金額とを通知している。このようにすることで、大部分のICカード利用者は、たとえ預金が0であろうとも、購入価格の10%が返済されるために、必ずICカードを最後にはSS協会に届けることになる。このICカードは単なるカードだけでなく、経済的

価値を有しているものであり、それ自体が通貨の代用として社会に流通し得るものでもある。ICカードを受け取ったSS協会での処理は上述と同様の処理を行う。

逆に、カードの現金化を行わず、次のカード又はそのカード自身に金額移算のみを許すようにすれば、カードの預金変換に対する防衛機構の重要性を、現金で精算する場合より低くできる。

従って、プログラム権利者は単にプログラムをユーザに貸した後は、所定の銀行に利用料金が振り込まれるのを待っていれば良い。更に、社会に若しデッド・コピーという不正を行うものがいれば、該コピープログラムには、オリジナルなプログラム権利者のプログラム権利者IDが入っており、その利用料金はすべて正当なプログラム権利者の口座に振り込まれることになり、「複写大歓迎」という有償プログラムシステムが実現する。

そして、他人のプログラムを複写して、自分のプログラム権利者コードを付与する不正な取引が存在しても、その履歴が把握可能となり、損害賠

償額が明確になり得るので、その点でもメリットがある。

第3図は、本発明の実施例であり、前図と同記号のものは、前図と同じものを示し、11aは通貨単位も含む利用可能金額を格納する預金メモリ、11bは利用明細メモリ11cへの書き込み位置を示すポインタ・メモリ、21はプログラム権利者ID21a、プログラム番号やその版数等のプログラム識別ID21b、その利用料金データ21c、利用状況や特約データ等で料金を決める割引率データ21d、等を格納するプログラム・データ・メモリ、22は割引処理部、23は預金があることをチェックする預金チェック処理部、24はポインタ・メモリの示す値が所定値以上か否かを判定する利用明細メモリ領域チェック処理部、25は預金メモリに格納されている預金でプログラムを利用できるか否かをチェックする利用可能性チェック処理部、26は預金メモリ更新処理部、27はポインタ・メモリの内容を+1するポインタ更新処理部、28は利用明細メモリ11cへの

書込み処理部。28はOSへのOS応答処理部。  
30は利用明細メモリ11cのアクセス位置を制  
御するアクセス制御部。31はユーザのチェック  
を行うユーザ・チェック処理部。32はキーボ  
ードよりの指示で残高メモリ11aへの増分をその  
妥当性をチェックした上でフェッチしてくるため  
の増分フェッチ処理部。33は残高メモリ11a  
への加算処理部。34はポイントが0より大か否  
かを検出するポイント値チェック処理部。35は  
利用プログラム履歴メモリ11の読み出し処理部。  
36は外部への伝送処理部。37はポイントを一  
1するポイント減算処理部である。

ソフトウェア・サービス・ユニットSSUは、  
3大処理に分けられる。その1は有償プログラ  
ムの利用時の処理。その2は残高メモリ11aの  
内容を増加する処理。その3は利用プログラム履歴  
メモリ11の内容を外部へ伝送するための伝送処  
理である。それらを順次説明してゆく。

キーボード16よりの指示でまず残高メモリ  
11aが残高チェック処理部28でチェックされ、

にしたりする制御も可能である。勿論、特約デー  
タの有無をチェック可能で、特約のない利用者は  
利用書固有データ中に利用特約が存在せず、その  
場合、「特約がありませんので、このプロ  
グラムは使用出来ません。」なるメッセージが表  
示装置17に表示される。更に、利用料金として、  
1より小さい比例定数を有する等比級数を採用す  
れば、無限回使用した場合の料金がわかるので、  
将来の必要経費が見通し可能となるような、利用  
者に喜ばれる方式をとることも可能となる。

この割引処理部の出力が得られ、そのプログラム  
を利用した時の料金を意味している。その値を用  
いて、利用可能性チェック処理部25は残高メモ  
リ11aの値から割引処理部の出力値を減じ、そ  
の結果が負か否かをチェックする。負であれば、  
「残高が不足しています。」なるメッセージを表  
示装置17に表示する。負でない時には、残高メ  
モリ更新処理部26が残高メモリ11aの値を該  
減じた結果に更新し、ポイント更新処理部27に  
て、ポイント・メモリ11bの内容を+1する。

負(特別な処理形態を考えることも可能だが本来  
あり得ない)または0の時は「残高無し」のメッ  
セージが表示装置17に表示される。それ以外の  
時、次にポイント・メモリ11bの内容がチェッ  
クされ、MAX値の超過有無が調べられる。MAX  
X値として、利用明細の記載済み項目数、又は、  
利用された金額でSS協会へ報告未完了の累計を  
用いることが可能である。MAX値を超過してい  
る時、「利用明細メモリが不足しています。一度、  
利用明細をSS協会へ転送して下さい。(又は、  
ICカードを差し代えて下さい。)」なるメッ  
セージを表示装置17に表示する。

ポイント・メモリ11bの値、即ち、記載済み  
項目数が所定のMAX値以内であれば、次に、プ  
ログラム・データ・メモリ21より利用料金デー  
タ21cと割引率データ21d及びユーザ識別符  
号メモリ9の情報をを用いて、割引処理部22は利  
用料金を求める。例えば、特定の資格を有する人  
は特別割引にしたりする処理であり、所定回数  
又は試用期間として格安にしたり、または無料

そして、書込み処理部28は利用明細メモリ11  
cに、ユーザ識別符号、プログラム権利者ID  
21a、プログラム識別ID21b、通貨単位を  
含めた料金等を格納する。ポイント・メモリ11  
bはアクセス制御部30に対して新格納場所をポ  
イントしており、その新格納場所へ上記データは  
格納されることになる。その後、OS応答処理部  
29はOS部13に対して利用可能を指示する。

尚、上記実施例では、残高メモリ11aは利用  
された分だけ直ぐに減じ、常に利用可能な金額を  
表示している例で説明したが、イニシャル値を格  
納して、その値を利用明細メモリの内容が外部へ  
出力されない限り変更せずにそのままにして置き、  
残高を問われた毎に、該イニシャル値から利用明  
細メモリにある明細の合計を差し引くという処理  
を行う形態にしてもよいのは勿論である。

次にその2としての処理を説明する。

キーボード16よりの指示でまずユーザ・チェ  
ック処理部31が動作する。ここでは、これから  
の処理を行う者が、正当であることをチェックす

る。(この個人履歴処理は、本説明では省略したが、その1の処理及びその8の処理で採用可能のものである。) 簡単に、キーボード16からの暗証コードとユーザ識別符号メモリ9の内容とが矛盾しないか否かをチェックする。矛盾していれば、「暗証コードを入れ直して下さい。」なる旨のメッセージを表示装置17に表示する。矛盾しなければ、入出力処理部12を介して増やすべき残高をフェッチする処理を行う。これは、オンラインでセンタより所定の暗号チェックを交わした後、キーボードから増やすべき残高を得てもよいし、入出力処理部12に接続されたICカードより得てもよい。

得られた増やすべき残高は加算処理部33にて残高メモリ11aの内容に加算され、再び残高メモリ11aに格納される。

次に、その3の処理について説明する。

キーボード16からの指示でまず、ポイント・メモリ11bの内容がポイント値チェック処理部34にてチェックされる。0より大きくない場合

は、「利用明細メモリの伝送は完了しました。」なるメッセージを表示装置17に表示する。0より大きい場合、読み出し処理部35は該ポイント・メモリ11bで指定されている領域の利用明細メモリ11cの内容を読み出し、伝送処理部36はその内容を入出力処理部12により、オンラインでセンタであるSS協会へ通知する。ICカードの場合であれば、このICカードへの記憶処理がなされる。そして、次にポイント値処理部37はポイント値を-1し、再び制御をポイント値チェック処理部34へ移す。ポイント値チェック処理部34は0より大きいか否かをチェックし、0より大きくない場合は、その旨を表示装置17に表示する。この様にして、ポイント・メモリ11bの内容が0になる迄処理は繰り返される。その結果として、利用明細メモリ11cの内容はすべてSS協会へ通知されることになる。

上記の様な構成にしたことにより、残高メモリ11aが0になる前でもSS協会へその利用明細メモリが通知されるので、金額が残高メモリ11

aに格納されても、利用明細への記入量は限定されるので、たとえ障害で利用明細が失われても、被害は小さく留めることができる。更に、ユーザ識別符号により、利用明細メモリ領域チェック処理部24でのMAX値を動的に変更する様にすれば、信用度の高いユーザに対しては、その利用明細の報告はより長期間単位とし、信用度の低いユーザはその利用明細の報告を短期間とする等の制御も自由に行うことも可能となる。

第4図は、本発明の他の実施例であり、図において、前図と同記号のものは前図と同じものを示し、37は主記憶装置、38は中央処理装置、39はチャネル装置、40は入出力制御装置、41はICカード・リーダー、42はICカード、43はマイクロ・プロセッサ、44はマイクロ・プロセッサ43を動作させる為のプログラムが格納されたEPROM(イー・ピー・ロム)、45は入出力ポート、46は内部バスである。

動作は、前図と全く同じであり、ただメモリが内部になく、外付けになっている点異なるだけ

である。従って、有償プログラムが利用可能か否かはICカード42内の残高メモリ11aを用いてチェックされ、OKならばマイナスの更新処理がなされ、且つその利用明細が利用明細メモリ11cに格納される。

但し、料金の支払いについては、まずユーザは所定のSS協会の代理店等で購入をする。購入の際の価格に応じた金額が残高メモリ11aに格納されている。従って、ユーザはそのICカード42を利用可能金額だけ利用すると、そのICカードをSS協会に届ける。すると、その残高メモリ11aに残高があればその金額と、利用した金額に対して10%の払い戻し金額とを合わせて得ることが可能となっている。故に、SS協会では、確実にプログラム権利者に対してその利用料金を支払うことが可能となる。

また、ICカードの中のEPROMに残高メモリ11aの内容を増加する方向の処理を行えない様なプログラムを格納したり、プログラム固有データが勝手に変更されないように該データを暗号

化してプログラム中に分散して配置し、ICカード中に設けたキーで該暗号を解くようなプログラムを格納したりしておけば、それだけICカードの不正使用は減ることになる。

尚、上記全文における利用者識別符号は、利用者が割引等の、利用者の身元に依存する特権を得たいとき、または1個のソフトウェア・サービス・ユニット(SSU)を用い、複数の支払元の利用者がプログラムを利用するとき以外には、必ずしも必要ではない。利用者識別符号が利用明細メモリ11cに格納されなければ、その利用者固有の割引等を受けられない反面、利用明細がSS協会に報告されてもその利用者が知らないので、利用者のプライバシーは完全に保たれる。

また、上記実施例では残高メモリが“0”の時には、処理装置が動作しない例を用いたが、残高が少なくなってきたらその旨の表示を出す方式や、所定金額迄は警告を表示しつつ消費を許し、該値の値を更に越えたらソフトウェア・サービス・ユニットの基本部分を破壊し、以後、ソフトウェアの

サービスを受けられない様になってしまう方式も考えられる。こうすることで、即停止の心配という心理的バリアを除くことが可能となる。

さらに、ソフトウェアを使用するたびに、コイン投入等、自動的に入金情報を残高として格納する方式も可能である。

また、ソフトウェア・サービス・ユニットは、通常のOSとデータ処理システムにおいて共存することも可能であり、既存システムにこのソフトウェア・サービス・ユニットを組み込むことが可能であるので、普及ということにも効果がある。

また、プログラム登録データが勝手に変更されることを防ぐために、有償プログラムを暗号化しておき、その暗号を解くためのキーをICカード中に備えることも可能である。

また、ICカードにおいても、その内容をSS協会に提出して再利用可能な如く構成することも、EPROMの使い方次第で可能である。

更に、本発明は、有償プログラムを例にして説明したが、ソフトウェアの有償サービス・例えば

電波による有償画像サービス、ビデオカセット、及び、有償情報提供等にももちろん適用可能である。

#### 例 発明の効果

以上述べた様に、本発明によれば、ソフトウェアの利用状況に関してソフトウェア権利者は確実かつ容易に把握することが可能となり、ソフトウェア権利者の権利の保護を図りつつ多様な形態でのソフトウェアの利用を促進することができる。

#### 4. 図面の簡単な説明

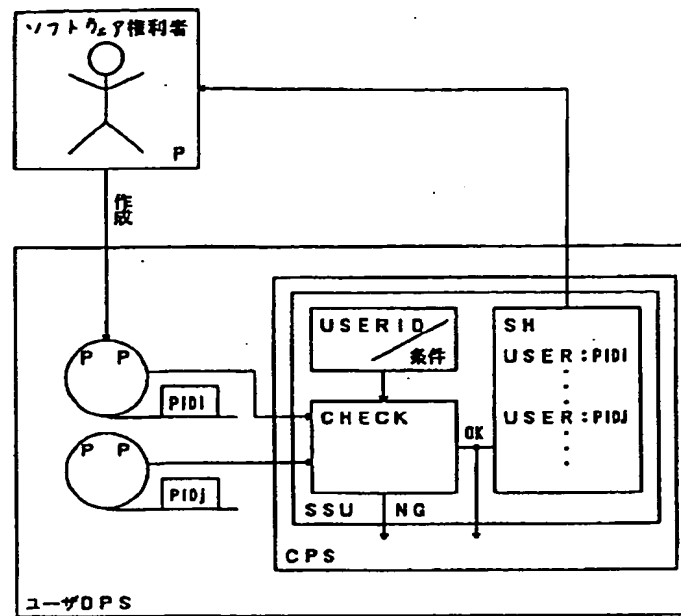
第1図は本発明の概念を説明するための図、第2図と第3図は本発明の実施例、第4図は本発明の他の実施例である。

図において、1はデータ処理システム、2はSS協会、3-1〜3-nはプログラム権利者、5は有償プログラム、8はソフトウェア・サービス・ユニット、9はユーザ識別符号メモリ、10は利用可否判定処理部、11は利用プログラム履歴

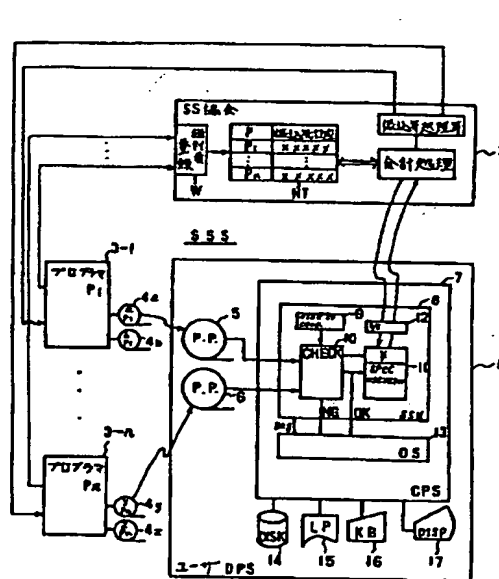
メモリ、11aは残高メモリ、11bはポインタ・メモリ、11cは利用明細メモリ、12は入出力処理部、16はキーボード、17は表示装置、21はプログラム・データ・メモリ、25は利用可能性チェック処理部である。

特許出願人 森 亮 一  
代 理 人 弁護士 長谷川 文廣(外1名)

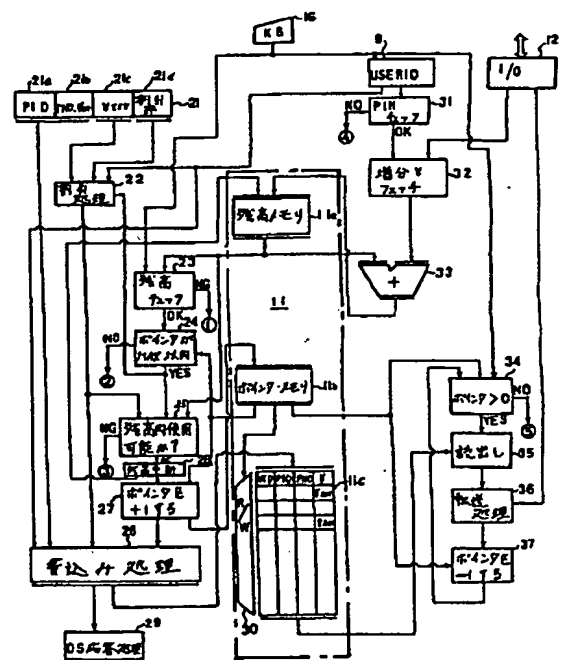




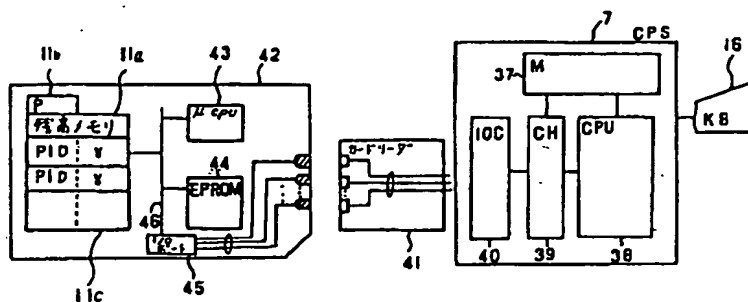
第 1 圖



第 2 回



३



第 4 図

(19) Japan Patent Office (JP)

(11) Patent Application Disclosure Number

(12) Laid Open Patent Publication (A)

Heisei 4-64129

(43) Laid Open: Heisei 4 (1992) February 28

(51) Int. Cl. <sup>5</sup>	Classification symbol	Office internal number
G 06 F 9/06	450 H	7927-5B
	450 L	7927-5B
12/14	320 E	7165-5B

Examination request: Done

Number of Inventions: 1 (Total 10 pages)

---

(54) Patent Title: Software Management System

(21)<sup>1</sup> Patent application No.: Heisei 2-211406

(22) Filing date: Showa 58 (1983) October 5

(61) Patent application No.: Continuation in Part of Showa 58-186100

(72) Inventor: Ryou-ichi Mori  
1-24-12 Hakusan, Bun-kyou-ku, Tokyo

(71) Applicant: Ryou-ichi Mori  
1-24-12 Hakusan, Bun-kyou-ku, Tokyo

(74) Agent: Patent Attorney: Hasegawa and another

---

Specification

1 Patent Title: Software Management System

2 What is claimed is:

Claim 1:

Software management system having

software storage means to store software-specific data corresponding to software programs,

user-specific data storage means to store user-specific data corresponding to software users,

a usage approval determination means to determine whether or not a user can use a software program, using at least a portion of each of said software-specific data and said user-specific data,

and a software usage history storage means to store data corresponding to the software-specific data when said usage approval determination means determines that a software program can be used,

wherein the software management system is characterized in that the stored content in said software usage history storage means is managed in accordance with said software-specific data.

#### Claim 2:

Software management system which is described in Claim 1, whereas the software management system of Claim 2 is characterized in that

it has a software right holder identification code as the software-specific data,

it has software usage conditions as user-specific data,

and it enables monitoring of the usage of a software program with respect to a software right holder by storing the software right holder identification code in the software usage history storage means when the usage approval determination means determines that the software program can be used.

### 3 Detailed Explanation of Invention

#### (a) Field of Industrial Application

The present invention relates to a management system of software such as computer programs and video software, which are intangible assets. In particular, it pertains to a software management system in which software usage can be monitored based on conditions such as for each software right holder.

(b) Background of Invention

In recent years, as data processing systems advance, various programs are being sold. However, the protection of such programs is not perfect and there are many occasions of unlawful usage of programs. In order to prevent such unlawful usage, new patent laws, copyright laws and furthermore, special laws are being studied. However, software has appeared into existence only in very recent years and any of these provisions is not yet perfect. This stems not only from the difference between tangible and intangible assets, but also from whether right holders are aware of transactions. Whether or not this point is successfully resolved will be the key in assuring the protection of software programs.

(c) Issues with Prior Art

Once software, for example, computer programs, is transferred to a user, right holders of the software can not know how much the programs are used. Hence, usage is not controlled at all. This condition means that payments would not be made for programs when these programs are purchased with an agreement that payments are to be made based on the amount of their usage. This is equivalent to allowing a sort of "dine and dash". In such a system, not only do usage fees of programs end up being very high, but also a deluge of crude software will be allowed and a desire to provide software which is accompanied with detailed services will be lost. Accordingly, if society provides a system in which program right holders of computer programs are protected with detailed attention, creation of better computer programs will be encouraged in said society. As a result, society will advance further. This is a very important issue in the future when computer programs will occupy significant importance.

There is a conventional method to protect against conventional unlawful actions of duplication including copying computer programs. In this method, a serial number is assigned to a piece of hardware such as a computer, a code corresponding to this serial number is assigned to a piece of software. When this program is executed, the serial number of the hardware and the serial number assigned to the software are compared. Only when they match, said software is allowed to be executed. Thus, copied programs can not be used. (Copied software does not have a matching serial number and can not be executed.)

However, managing serial numbers is a cumbersome task. Moreover, when software is placed at stores for sales, it is impossible, in reality, to implement this method which uses serial numbers.

(d) Purpose of Invention

Accordingly, the present invention has a purpose to enable right holders to monitor how much software is being used.

(e) Design of Invention

This purpose is achieved by a software management system having software storage means to store software-specific data corresponding to software programs, user-specific data storage means to store user-specific data corresponding to software users, a usage approval determination means to determine whether or not a user can use a software program, using at least a portion of each of said software-specific data and said user-specific data, and a software usage history storage means to store data corresponding to the software-specific data when said usage approval determination means determines that a software program can be used, wherein the stored content in said software usage history storage means is managed in accordance with said software-specific data.

(f) Embodiment

Figure 1 explains the concept of the Software Service System (SSS) of the present invention. In the figure, P is a software right holder, PP is a piece of software with a price (Program Product), PIDi is a software-specific data, USERID is a user-specific data, CHECH is a usage approval determination means, SH is a software usage history storage means, DPS is a data processing system, and SSU is a software service unit.

Software right holder P provides a piece of software with a price PP including its software-specific data PIDi. The user's DPS is designed so that the software passes through its software service unit SSU when software with a price PP is used. Software service unit SSU contains user-specific data USERID. When a usage request is made for software with a price PP, usage approval determination means CHECH checks the usability of the requested software, using its user-specific data USERID. For example, it checks whether the user is authorized to use the program, whether the usage fee is within the usage limit allowed for the user, whether the usage is within the allowed time limit, and so forth. If it determines that the usage is allowed, a message to this effect is notified to the operating system (hereafter, referred to as OS), which is not shown in the figure, of data processing system DPS. At the same time, user-specific data USERID and software-specific data PIDi are stored in software

usage history storage means SH. If the usage is not allowed, then the response to that effect is sent to the OS of data processing system DPS.

The content of software usage history storage means SH can be verified by software right holders P. At the same time, for example, software usage fees can be billed for each of the software right holders based on the content of software usage history storage means SH.

The present invention enables software right holder P to monitor how much software with a price PP is used. Therefore, usage fees can be collected in accordance with the usage. By holding the purchase price of software low, users can test various pieces of software and can continue to use good pieces of software. Thus, crude software will eventually not be used. Good software will be used frequently. Therefore, software right holders P will compete to provide better software.

Figure 2 shows an example in which the present invention is applied. In the figure, 1 is a user's data processing system, 2 is an SS organization, which a management organization uses to manage programs with a price as an example of software with a price, 3-1 ~ 3-n are program right holders, who have the right to receive compensation when his/her program with a price is executed (when there are a plurality of program right holders for one program, compensation distribution ratios are also stored), 4a ~ 4z are programs with a price, wherein 4a and 4b are created by program right holder 3-1 and 4y and 4z are created by program right holder 3-n, 5 and 6 are programs with a price, 7 is a central processing system, 8 is a software service unit (SSU), 9 is a user identification code memory, which is a portion of a user-specific data storage means, to store a user identification code, 10 is a usage approval determination means to determine whether or not a program with a price can be used, 11 is program usage history memory, which is the other portion of user-specific data storage means, including a usable amount memory to store information including a currency symbol (such as \$, yen and pound) and a software usage history storage means, 12 is an input/output processing section to process input and output of the content of program usage history memory 11, 13 is an OS section in central processing system 7, 14 is a magnetic disk device, 15 is a printer, 16 is a keyboard, and 17 is a display device.

In the system shown in Figure 2, when a program right holder wishes to create a program, allow general public to use the program as a program with a price, and to ensure collection of usage fees, first the program right holder goes to the SS organization. There, the program right holder registers his/her receiving account name for the usage fees and receives a program right holder identification code (hereafter, abbreviated as a program right holder ID). The organization issues a

different program right holder ID to each program right holder. This process is performed, for example, at a clerk window W in Figure 2 and is managed by management log NT. As an example, P1 is assigned to program right holder 3-1 as his/her program right holder ID.

Let us assume that program right holder P1 creates programs with a price 4a and 4b. The program right holder attaches program-specific data, such as said program right holder ID P1, a program version number and a usage fee, to these programs. The programs are stored in a storage media such as a cassette magnetic tape and are sold at stores or through direct mail. Of course, they can be sold on-line over a communications network.

On the other hand, users seek programs, which are suitable for their usage, at stores or in magazines. A user finds that a program he or she is seeking is created by the members of the SS organization. Let us assume that this program is 4a in Figure 4a. The user can obtain this program at a very low price or for free. Software service unit 8 must be installed in user's data processing system 1. The software service unit is a mechanism to process programs which are registered at SS organization 2.

Software service unit 8 has usage approval determination means 10 and input output processing section 12. Usage approval determination means 10 checks the data from user identification code memory 9, the program-specific data attached to the program with a price to be used, and the content of the program usage history memory, and determines whether or not a program can be used. Input output processing section 12 outputs the program usage history from the content of program usage history memory 11 and updates the usable amount. Input output processing section 12 can be implemented in a different form by each user. For example, it could transfer to a remote file on-line. Or, it could be an external storage media such as a magnetic card or an IC card containing a microprocessor. Depending on its usage form, a variety of software service units 8 can exist.

When data processing system 1 executes program with a price 5, in other words, when OS section 14 detects the usage of said software from the interpretation result of the job control language, a message to this effect is notified to software service unit 8. This may be a control which is comparable to a special interruption. Software service unit 8 reads the usable amount in program usage history memory 11. At the same time, software service unit 8 obtains a usage fee from the program-specific data in said program with a price 5. Next, a process is performed to deduct the usage fee from said usable amount. If this result is negative, the program can not be used. A message to this effect is sent to OS section 13. For example, a system message, "Program 4a can not be used because



of a deficient balance”, will be displayed on display device 17. (As mentioned afterwards, a conditional usage may be allowed even with a negative result.)

Conversely, if the result of deducting a usage fee from said usable amount is not negative, said result of deduction is stored in the storage area for a usable amount in the program usage program history memory. At the same time, program right holder ID P1 attached to the program, the usage fee and the user identification code in the user identification code memory are stored in program usage history memory 11 as the usage detail. Of course, the code to identify the program may be an unnecessary piece of information for a fee payment. However, it has a value in monitoring the usage of the program. Thence, it is stored in program usage history memory 11. Then, a request is made to OS section 13 to continue the process.

Thereafter, similar processes are undertaken. Hence, program usage history memory 11 contains the balance, in other words, the usable amount. At the same time, information is stored as to how much each of the users identified by his/her user identification code must pay to each of the program right holder IDs.

In the present system, the area of program usage history memory 11 stores a certain limit amount or quantity. The present system is designed so that when said limit is exceeded, a process is necessary to empty out the contents of program usage history memory 11. In a case such as a large limit amount, a means is provided to send an intermediate report of the usage to the SS organization. The content of the balance memory in program usage history memory 11 can be externally increased through the input output processing section. A similar design to a bank-on-line system can be implemented. Or a design can be implemented in which said IC card is purchased from the SS organization for a certain price and the balance is increased by the fee amount stored in the IC card.

In the present invention, it is possible to include special contract data as part of the program-specific data. In such a case, said usage approval determination means 10 determines that the program having the special contract data in its program-specific data can be used by users having user-specific data which contains a special contract usage code. This arrangement is convenient when program right holders want to monitor users for specific programs. In order for users to obtain a special contract usage code, they need to have a special contract with the SS organization. In order to have a special contract, information such as a user name is registered at the SS organization. Hence, it is easier for program right holders to monitor information such as names. Moreover, making it obligatory in the special contract to report such information makes it possible to prevent a dumping-like act such as an

unlawful act for competition by choosing not to pay a programmer, rather than gaining through a discount by reporting.

In the present system, it is necessary for the SS organization to know the usage details in program usage history memory 11 in order to pay to the program right holder of a certain program that is used. As a method to realize this, a means is implemented so that, for example, 10% of the fee is paid back to a user when the user reports the usage details. Accordingly, a user can gain if he/she reports. Thence, most users will report and program right holders can guarantee a collection of fees for the usage. In Figure 2, SS organization 2 receives a report, references management log.NT using program right holder ID, and processes a remittance of a fee to a corresponding receiving account.

The same is done when an IC card is used. It is conceived that used IC cards are brought to the SS organization. IC cards can be brought to the SS organization in order to exchange them for cash. In other words, the SS organization determines a reward for returning IC cards based on the usage amount to those who return IC cards. For example, 10% of the accumulated usage amount and the balance of the usable amount are paid back. By doing so, most IC card users will certainly return used IC cards to the SS organization even with a zero balance, because 10% of the purchase amount will be paid back. These IC cards are not just physical cards, but also carry a monetary value. They can be circulated within society as a replacement for currency. After receiving IC cards, the SS organization performs the same process as described above.

Conversely, instead of cashing in the cards, a monetary amount can be transferred to new cards or returned cards. Then, the importance of protection mechanism against tampering with the balance becomes lower than when the cards are cashed.

Accordingly, program right holders merely have to sit and wait until usage fees are deposited to certain banks after giving programs to users. Moreover, if society has people who perform unlawful acts of making a dead copy, copied programs carry a program right holder ID of the original program right holders. Thence, the usage fees of the copied programs will be remitted to the account of the rightful program right holders. Hereby, a system with programs having a price welcomes “making copies”.

Moreover, even if there are unlawful transactions in which one's own program right holder code is attached to counterfeit programs, the history can be monitored and the damage amount becomes clear. There is an advantage from this aspect as well.

Figure 3 shows another example of the present invention. Items with the same numeric labels as in the previous figures are the same as those in this figure. In Figure 3, 11a is a balance memory to store a usable amount including the currency unit, 11b is a pointer memory which indicates the writing position in usage detail memory 11c, 21 is a program data memory to store information such as program right holder ID 21a, program identification ID 21b such as a program number and version number, usage fee data 21c and discount data 21d which determines fees based on the usage and special contract data, 22 is a discount processing section, 23 a balance check processing section to check whether there is a balance or not, 24 is a usage detail memory area check processing section to determine whether or not the amount pointed by the pointer memory is equal to or more than a certain value, 25 is a usability check processing section to check whether or not a program can be used based on the balance stored in the balance memory, 26 is a balance memory update processing section, 27 is a pointer update processing section to increment the pointer memory, 28 is a write processing section for usage detail memory 11c, 29 is an OS response processing section to the OS, 30 is an access control section to control the access position in usage detail memory 11c, 31 is a user check processing section to check a user, 32 is an increase amount fetch processing section to check the validity of an increase amount to balance memory 11a and fetch the amount by the direction entered on the keyboard, 33 is an addition processing section for balance memory 11a, 34 is a pointer value check processing section to detect whether or not the pointer is larger than 0, 35 is a read processing section for program usage history memory 11, 36 is a transfer processing section to transfer to outside, and 37 is a pointer deduction processing section to decrement the pointer.

The software service unit SSU is divided into 3 major processes. The first is a process when a program with a price is used. The second is a process to increase the content of balance memory 11a. The third is a transfer process to transfer the content of program usage history memory 11 to outside. These are explained one by one.

Based on the direction entered on keyboard 16, balance check processing section 23 checks balance memory 11a. When the balance is negative (a special processing format can be considered but in principle, it is not possible) or 0, a message "no balance" is displayed on display device 17. Otherwise, the content of pointer memory 11b is checked in the next step. It is checked whether or not a MAX value is exceeded. As the MAX value, the number of items which are written into the usage details or cumulative used amount which are not reported to the SS organization can be used. When the MAX value is exceeded, a message "The usage detail memory has run out of space. Please transfer

the usage details to the SS organization. (Or, please insert another IC card.)” is displayed on display device 17.

If the value in pointer memory 11b, in other words, the number of written items, is within the MAX value, then the discount processing section 22 determines a usage fee based on usage fee data 21c and discount data 21d in program data memory 21 and the information in user identification code memory 9. For example, in this process, a person having a special status can receive a special discount. Another possible control example is that a very low fee or no fee can be applied at the beginning for a certain number of trials. Of course, it can be checked whether or not the special contract data exists. A user without a special contract does not carry a special contract data in his/her user-specific data. In such a case, a message “You have not completed a special contract. Hence, you can not use this program.” is displayed on display device 17. Moreover, a geometrical progression having a geometric ratio of smaller than 1 is employed for calculating a usage fee, then a fee can be obtained for unlimited usage. Then, a method can be implemented which can be appreciated by users because future expenses can be projected.

An output of this discount processing section is the fee for using the program. Using this value, usability check processing section 25 deducts the output value of the discount processing section from the value in balance memory 11a. It then checks whether or not the result is negative. If the result is negative, a message “There is not enough balance.” is displayed on display device 17. If it is not negative, then balance memory update processing section 26 updates the content of balance memory 11a to said deduction result, and pointer update processing section 27 increments pointer memory 11b. Then, write processing section 28 stores information such as a user identification code, program right holder ID 21a, program identification ID 21b, and the usage fee including the currency unit in usage detail memory 11c. The content of pointer memory 11b points to a new storage location for access control section 30. The above data is stored in the new storage location. Thereafter, OS response processing section 29 indicates to OS section 13 that the program can be used.

In the above example, it is explained that the content of balance memory 11a is immediately reduced by the usage amount so that it always shows the usable amount. Of course, an example can take a form in which balance memory 11a contains the initial value without revision until the content is output to the outside and whenever the balance is requested, a process is performed to deduct the total detail in usage detail memory from said initial value.

Next, the second process is explained.

The direction entered on keyboard 16 operates user check processing section 31. Here, it is checked whether a person who is about to initiate a process is a valid user. (This personal identification process can be employed in the first processor and in the third process, although the explanation is omitted.) Simply put, it checks whether or not a PIN code entered on keyboard 16 and the content in user identification code memory 9 contradict each other. If they contradict each other, a message "Please re-enter your PIN code." is displayed on display device 17. If they do not contradict each other, a process to fetch an increase amount to the balance is performed through input/output processing section 12. After a certain PIN check is performed by the center on-line, an increase amount to the balance may be entered on the keyboard, or from an IC card connected to input/output processing section 12.

Addition processing section 33 adds the obtained increase amount to the balance of the content of balance memory 11a and the result is stored back in balance memory 11a.

Next, the third process is explained.

First, the direction entered on keyboard 16, pointer value check processing section 34 checks the content of pointer memory 11b. When the content is not larger than 0, a message "The transfer of usage detail memory is complete." is displayed on display device 17. When it is larger than 0, read processing section 35 reads the content in the area of usage detail memory 11c which is indicated by said pointer memory 11b. Transfer processing section 36 transfers the content through input/output processing section 12 to an on-line center, which is the SS organization. When an IC card is used, a writing process to this IC card is performed. Pointer deduction processing section 37 decrements the pointer and returns the control to pointer value check processing section 34. The pointer value check processing section checks whether the content is larger than 0 or not. If it is not larger than 0, a message to this effect is displayed on display device 17. In this manner, the process is repeated until the content of pointer memory 11b becomes 0. As a result of this repetition, the entire content of usage detail memory 11c is notified to the SS organization.

Because the system is designed as above, the content of usage detail memory 11c is known by the SS organization even before the content of balance memory 11a becomes 0. Therefore, even if an amount is stored in balance memory 11a, the data amount which is written as usage details can be limited. Hence, even if usage details are lost due to a problem, the damage is small. Moreover, a user identification code can be used to actively alter the MAX value in usage detail memory area check processing section 24, a system can be implemented so that a reliable user can be required to report

his/her usage details once in a long time period while a less reliable user reports his/her usage details once in a short time period.

Figure 4 shows yet another example of the present invention. Items with the same numeric labels as in the previous figures are the same as those in this figure. In Figure 4, 37 is a main storage device, 38 is a central processing device, 39 is a channel device, 40 is an input/output control device, 41 is an IC card reader, 42 is an IC card, 43 is a microprocessor, 44 is an EPROM in which programs to operate microprocessor 43 are stored, 45 is an input/output port, and 46 is an internal bus.

The operation is the same as that in the previous figure. The only difference is that the memories are not internal, but external. Accordingly, balance memory 11a within IC card 42 is used for checking whether or not a program with a price can be used. If the checking result is fine, then the deducting update process is performed and the usage details are stored in usage detail memory 11c.

As for fee payment, a user performs a purchase at a representative store of the SS organization. An amount corresponding to the purchase amount is stored in balance memory 11a. Therefore, the user returns IC card 42 to SS organization after using it for the usable amount. Then, the user can obtain the balance, if any, in balance memory 11a and the 10% rebate of the used amount. Hence, the SS organization can be certain that usage fees will be paid to program right holders.

In addition, unlawful usage can be reduced by the following methods. A program can be stored in the EPROM in the IC card, wherein the program prevents a process to increase the content of balance memory 11a. A program can be stored so that program-specific data can be encrypted so that it can not be altered without permission and scattered within the program, so that a key in the IC card is used to decrypt the data.

A user identification code in the entirety of the above text is not necessarily needed except in the following cases. One is when a user wishes to obtain special rights which depend on the identification of the user. The other is when a plurality of paid users use a program on one software service unit (SSU). If a user identification code is not stored in usage detail memory 11c, a discount associated with the user identification code can not be obtained. However, a user identity will not be known to the SS organization when usage details are notified to the SS organization. Hence, the privacy of users can be completely protected.

Furthermore, when the content of the balance memory is "0" in the above examples, the processing devices do not function. However, other methods can also be conceived. For example, when the balance becomes small, a message to that effect can be displayed in one method. In another

method, down to a certain negative value, a negative value can be allowed while a warning message appears. When said certain negative value is exceeded, the core section of the software service unit is destroyed and thereafter, a user can not receive further software service in this method. Thus, a psychological barrier against an immediate stoppage can be eliminated.

Moreover, a method is also possible in which each time software is used, money such as coins is inserted and automatically the deposited amount is stored as balance.

Furthermore, a software service unit can coexist with a normal OS in a data processing system. In this method, a software service unit can be incorporated into an existing system. Therefore, this system may have an effect that it can be widely accepted.

In addition, in order to prevent alterations of program registered data without permission, it is possible to encrypt programs with a price and to include a decryption key within an IC card.

It is also possible to design the system so that an IC card can be reused after the content is submitted to the SS organization. This can be implemented by using the EPROM in a certain way.

Moreover, examples of the present invention are explained for programs with a price. Of course, the present invention is applicable, for example, to paid software services such as paid image services via radio wave, video cassettes, and paid information provision.

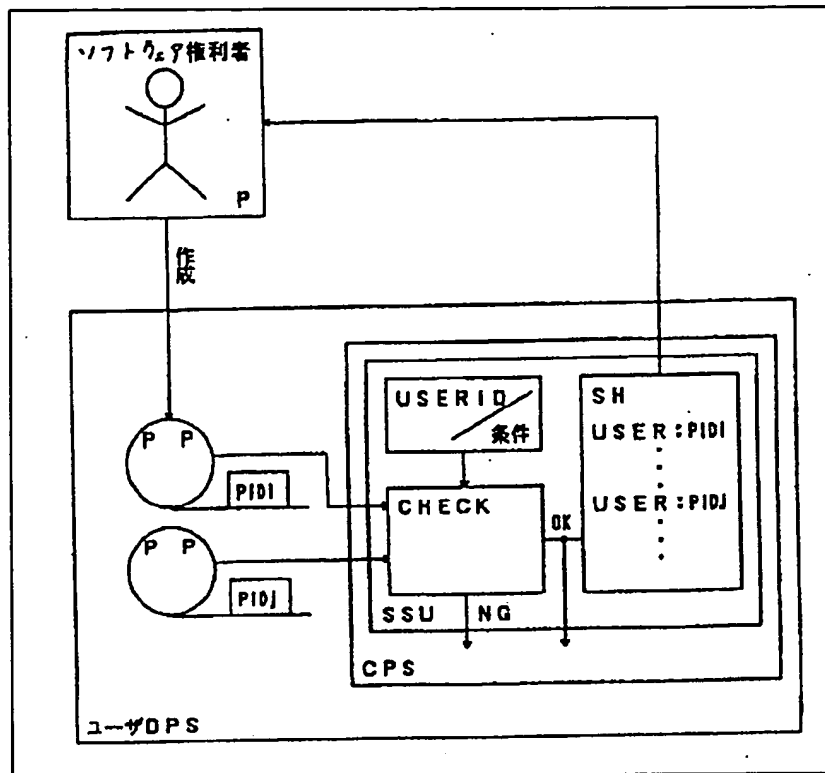
#### (g) Advantages of the Present Invention

As explained above, the present invention enables software right holders to ensure an easy monitoring of software usage. Hence, the rights of software right holders can be protected and at the same time a wide variety of software usage formats can be promoted.

#### 4 Brief Explanation of Figures

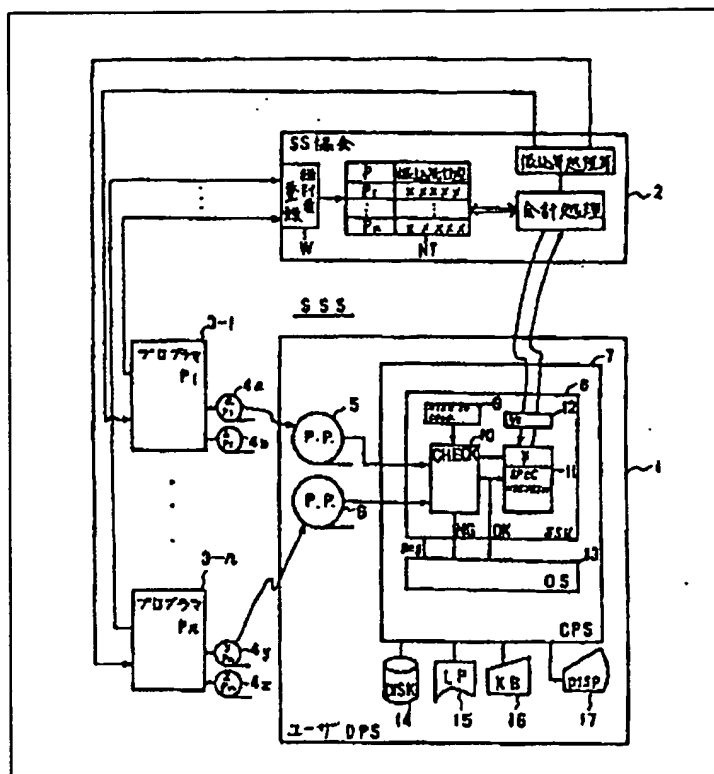
Figure 1 explains the concept of the present invention. Figures 2 and 3 show an example of the present invention. Figure 4 shows another example of the present invention.

Figure 1



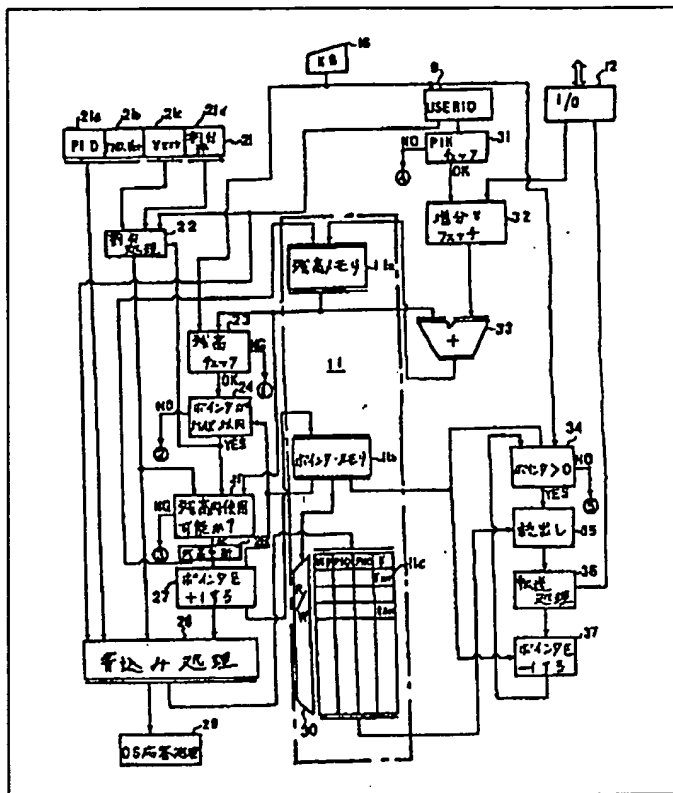
SSS	Software Service System	P	software right holder
PP	software with a price (Program Product)	PIDi	software-specific data
USERID	user-specific data		
CHECH	usage approval determination means		
SH	software usage history storage means		
DPS	data processing system	SSU	software service unit
W	Clerk window	NT	management log





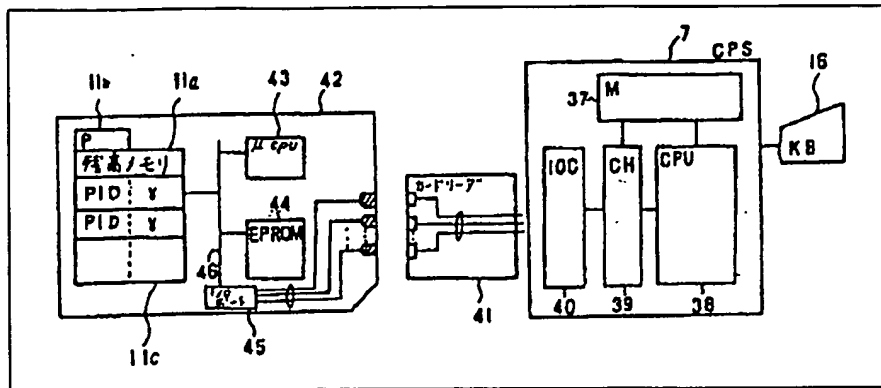
- |           |   |         |                             |
|-----------|---|---------|-----------------------------|
| 1         | user's data processing system   | 2       | SS organization             |
| 3-1 ~ 3-n | program right holders   | 4a ~ 4z | programs with a price       |
| 4a and 4b | created by program right holder 3-1   |         |                             |
| 4y and 4z | created by program right holder 3-n   | 5 and 6 | programs with a price       |
| 7         | central processing system   | 8       | software service unit (SSU) |
| 9         | user identification code memory, which is a portion of a user-specific data storage means,  |         |                             |
| 10        | usage approval determination means  |         |                             |
| 11        | program usage history memory, which is the other portion of user-specific data storage means, including a usable amount memory and a software usage history storage means |         |                             |
| 12        | input/output processing section   |         |                             |
| 13        | OS section in central processing system 7   | 14      | magnetic disk device        |
| 15        | printer   | 16      | keyboard                    |
| 17        | display device  |         |                             |

\_\_\_\_\_



- |     |   |     |                             |
|-----|---|-----|-----------------------------|
| 11a | balance memory                                    | 11b | a pointer memory            |
| 11c | usage detail memory                               | 21  | program data memory         |
| 21a | program right holder ID                           | 21b | program identification ID   |
| 21c | usage fee data                                    | 21d | discount data               |
| 22  | discount processing section                       |     |                             |
| 23  | balance check processing section                  |     |                             |
| 24  | usage detail memory area check processing section |     |                             |
| 25  | usability check processing section                |     |                             |
| 26  | balance memory update processing section          |     |                             |
| 27  | pointer update processing section                 | 28  | write processing section    |
| 29  | OS response processing section                    | 30  | access control section      |
| 31  | user check processing section                     |     |                             |
| 32  | increase amount fetch processing section          | 33  | addition processing section |
| 34  | pointer value check processing section            | 35  | read processing section     |
| 36  | transfer processing section                       |     |                             |
| 37  | pointer deduction processing section              |     |                             |

Figure 4



37	main storage device	38	central processing device
39	channel device	40	input/output control device
41	an IC card reader	42	IC card
43	microprocessor	44	EPR0M
45	input/output port	46	internal bus